

打破职能局限

科大讯飞百亿级交互业务的缺陷预防之路

孙玉

科大讯飞AI营销业务群测试架构师

个人简介-孙玉

2009 | 通信工程

安徽大学

2016 | 软件测试

科大讯飞

TERADYNE

2013 | 芯片测试

目录

1 A.I. 营销测试团队：挑战带来改变

2 缺陷预防引入的必要性

3 缺陷预防的应用阶段

4 七大缺陷预防实践

5 总结展望

1

第1部分

A.I.营销测试团队：挑战带来改变

人工智能国家队

2017年11月，科技部公布首批国家新一代人工智能开放创新平台



Tencent 腾讯



全球最聪明公司



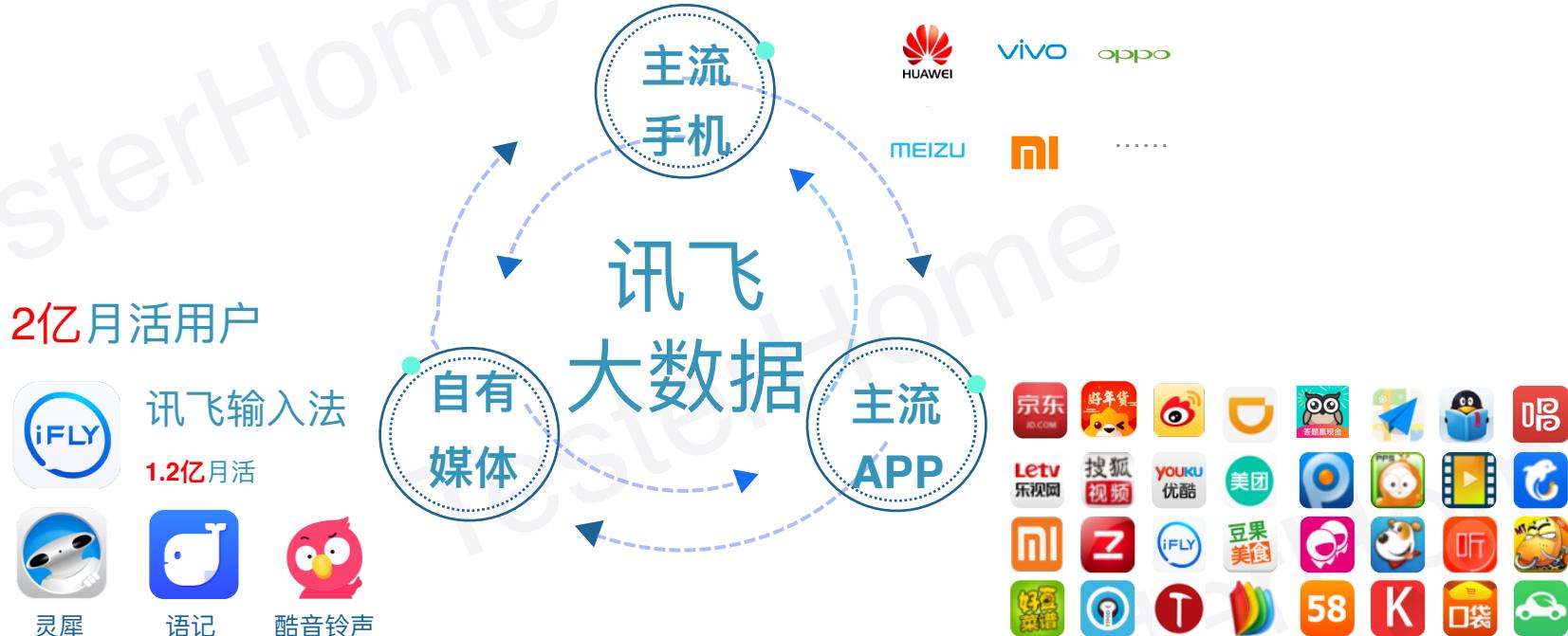
《MIT科技评论》

全球50大最聪明公司

科大讯飞被评为

中国第一，全球第六

A.I.营销测试团队：挑战带来改变



A.I.营销测试团队：挑战带来改变



A.I. 营销测试团队：挑战带来改变

语音合成

Nike《马拉松飞跑》



方言测评

雪佛兰《足球基音》



声纹识别

京东《新年说》

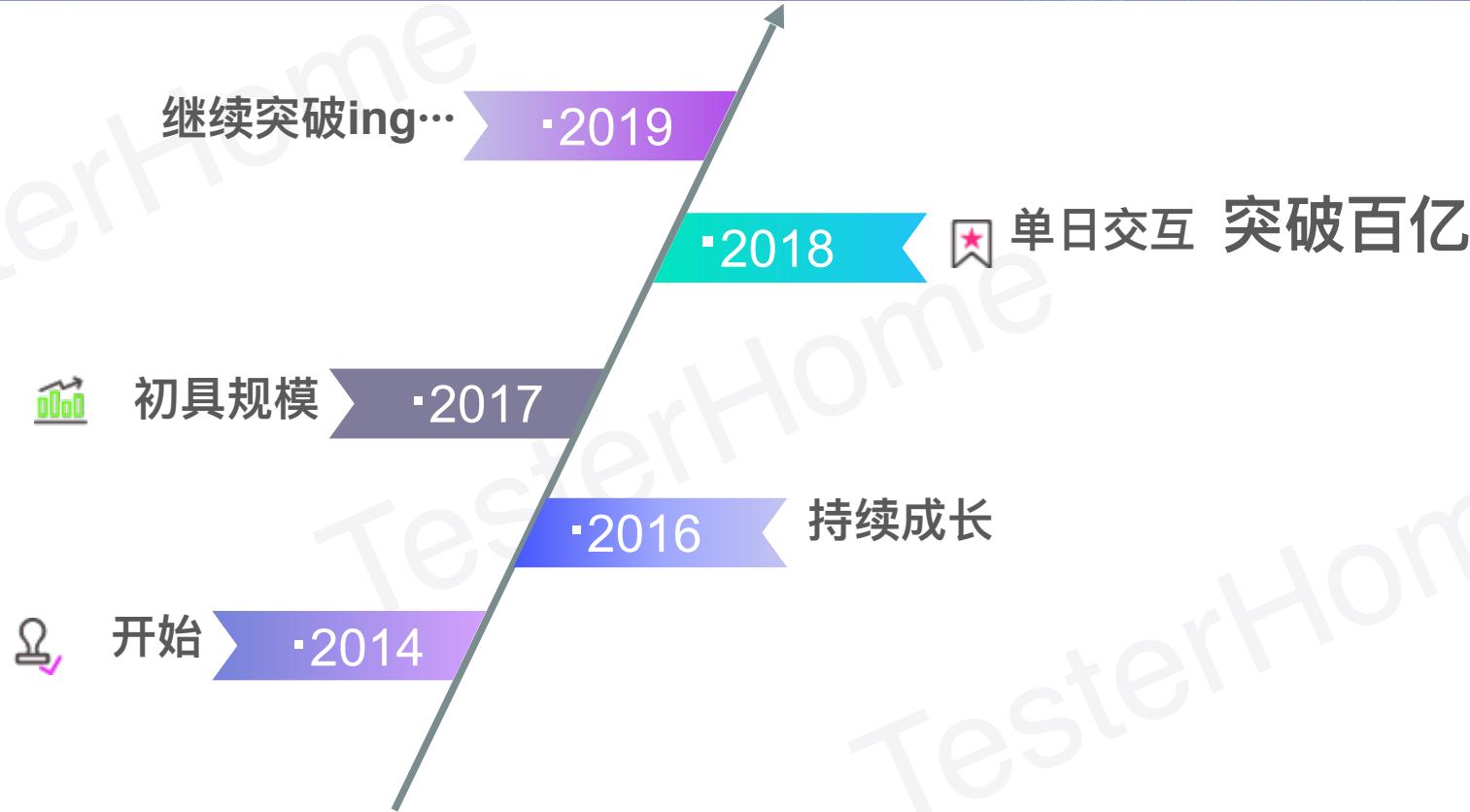


语音识别

唯品会

《12333.....》







起点



目标



关注重要节点 周密保障支撑 测试团队助力讯飞 AI 营销平台“双 11”再创佳绩

在 2018 年“双十一金秋攻略”战役中，讯飞 AI 营销平台再次取得销售大捷。佳绩的背后是平台全体同仁众志成城的拼搏和付出，其中，来自技术中心测试三部 AI 营销平台测试组的零故障后防技术支撑，也是重要一环。

一年一度的“双 11”，是 AI 营销平台各职能团队全年最紧张的时段。为全力保障“双 11”期间的业务功能演进质量，在“双 11”前期，技术中心派驻在 AI 营销平台的测试团队根据业务侧的演进目标，制定和落实了“双 11”期间的测试保障计划方案。通过识别重要节点、重点关注对象、重点版本，强化落实全流程质量风险防控的关键措施，明确责任落地，有效保障了需求响应的高效敏捷。“双 11”战役期间，迭代测试上线累计 46 次，无任何线上问题引入，为销售再创新高提供了稳固的后援技术保障。

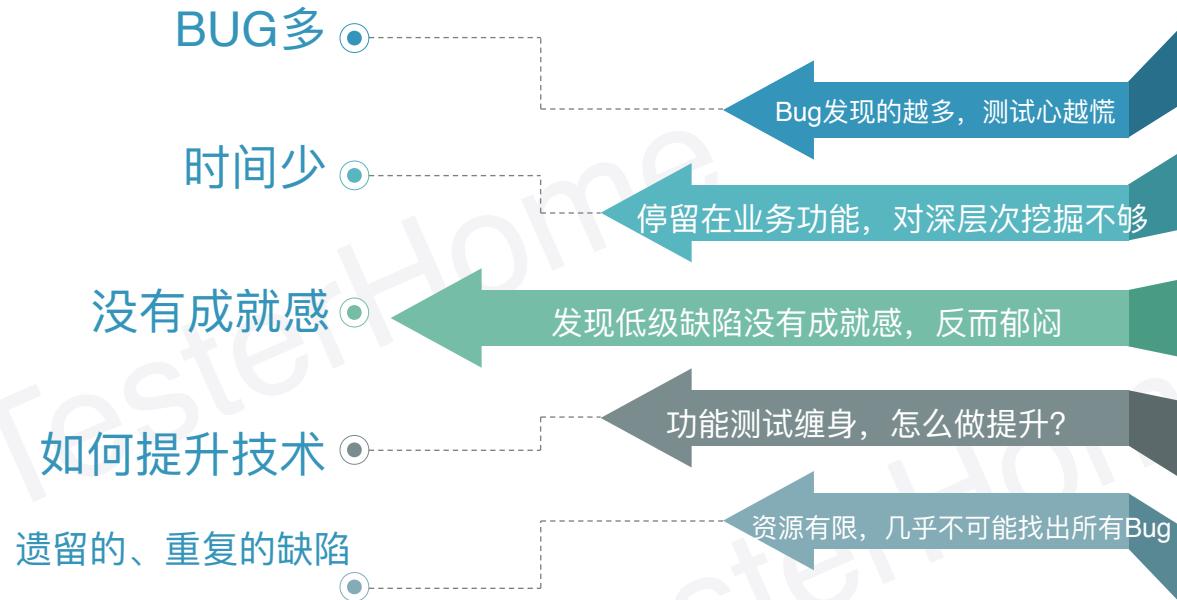
2

第二部分

缺陷预防引入的必要性

缺陷预防引入的必要性

测试同学每日烦恼



缺陷预防引入的必要性

需求返工

一改再改，牵一发而动全身。 . . .

风险不可控

隐藏的线上问题，拷问我们的灵魂！



功能不符合预期

上线前都觉得完美，上线后用户say No！

Delay

说好的需求，为什么没有兑现？

缺陷预防引入的必要性

惨痛的教训

我们做

用户反馈的：

反复修改、的讨论

历时2个月

不好用

8次版本构建

多次协作摩擦

不是想要的

投入人日=180+

4次上线

质疑：做这么久，就这样么？

其他吐槽…

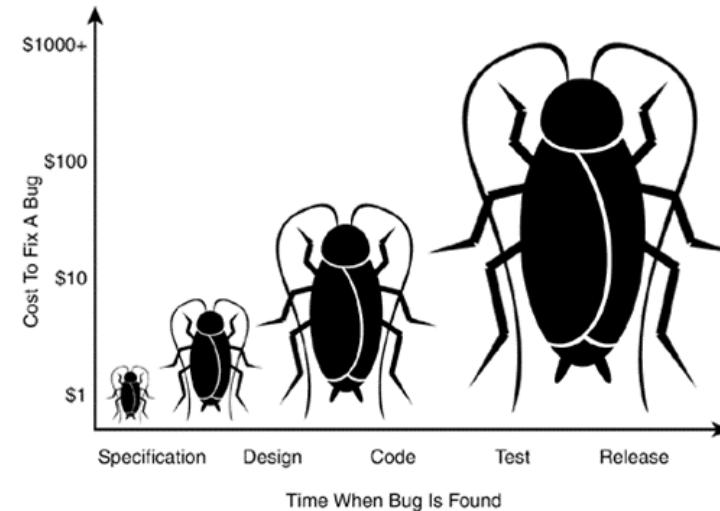
3

第三部分

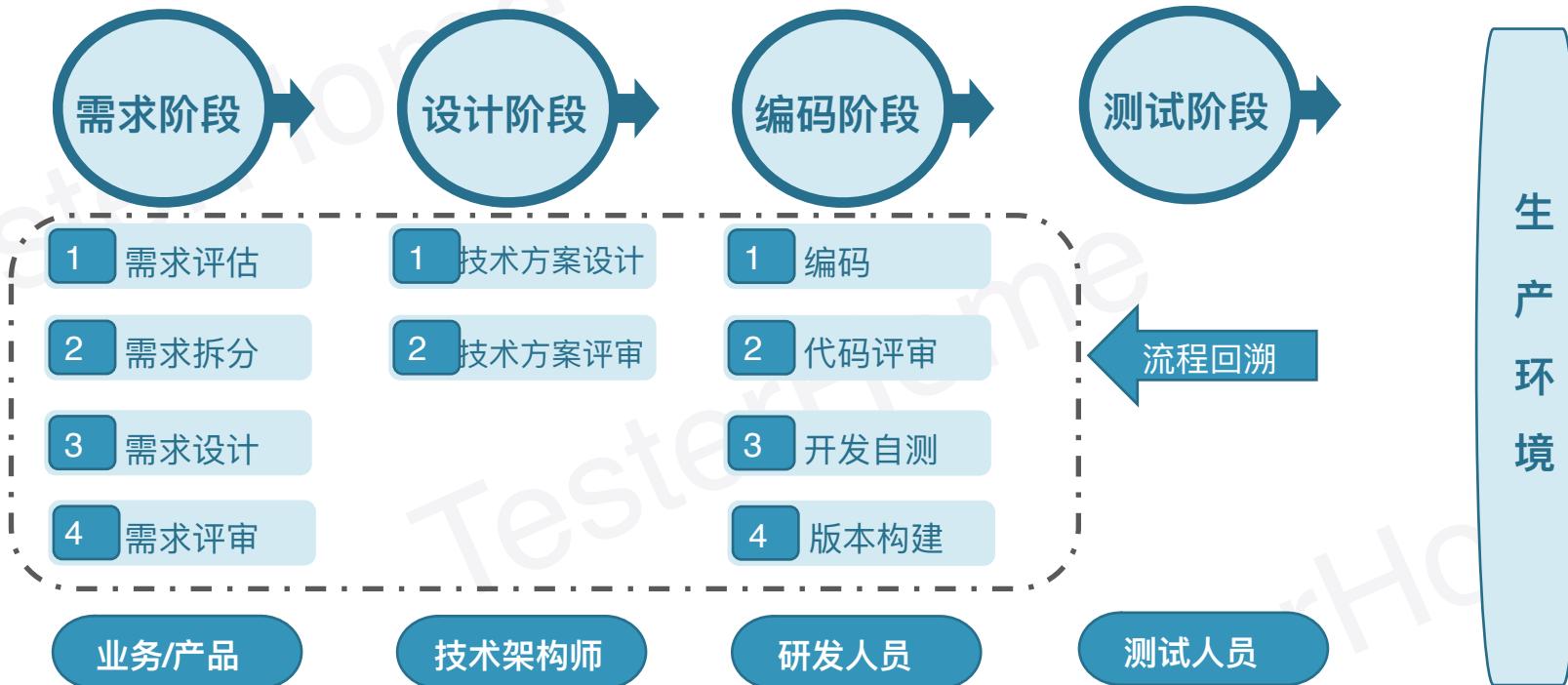
缺陷预防的应用阶段

缺陷预防的应用阶段

缺陷发现的越早，修复成本越低。



缺陷预防的应用阶段



明确缺陷注入的可能阶段!

4

第四部分

七大缺陷预防实践

缺陷预防实践一：业务演进马拉松之需求解耦



- 1 需求评估
 - 2 需求拆分
 - 3 需求设计
 - 4 需求评审
- 业务/产品

全都要 普适 大而全



把对产品的预期放在需求上



- ✗ 预估时间 不准确
- ✗ 需求变更 影响范围大
- ✗ 更容易 考虑不周



项目实践：

实践环节：需求拆分

落实人员：产品人员



测试人员：推进+帮助

主动推进



需求舍弃

敢于否定

确认粒度



✓ 小
✓ 轻
✓ 快
✓ 高



科大讯飞
iFLYTEK

TesterHome

实践效果：

The screenshot shows a Jira issue page titled "[产品演进] 升级XX模块，改造广告位模块样式及逻辑，支持..." (Upgrading XX module, refactoring advertisement module style and logic, supporting...). The page includes tabs for '编辑' (Edit), '备注' (Notes), '工作日志' (Work Log), and '更多...' (More...). A large teal callout bubble on the right says 'jira需求分解' (Jira requirement decomposition). The main content is a list of tasks under '子任务' (Sub-tasks):

1. 媒体模块：广告位模块样式改造
2. 开发后台：广告位模块和WAP流量注册
3. 多样屏：适配不同显示类型及尺寸
4. 【资源同】：将广告位模块拆分到广告云
5. 【广告位】：将广告位模块拆分到广告云
6. 广告位模块由模块化
7. 产品需求：广告位模块参数：广告位名称、广告位尺寸、广告正文
8. 【广告位】：广告位时的默认尺寸

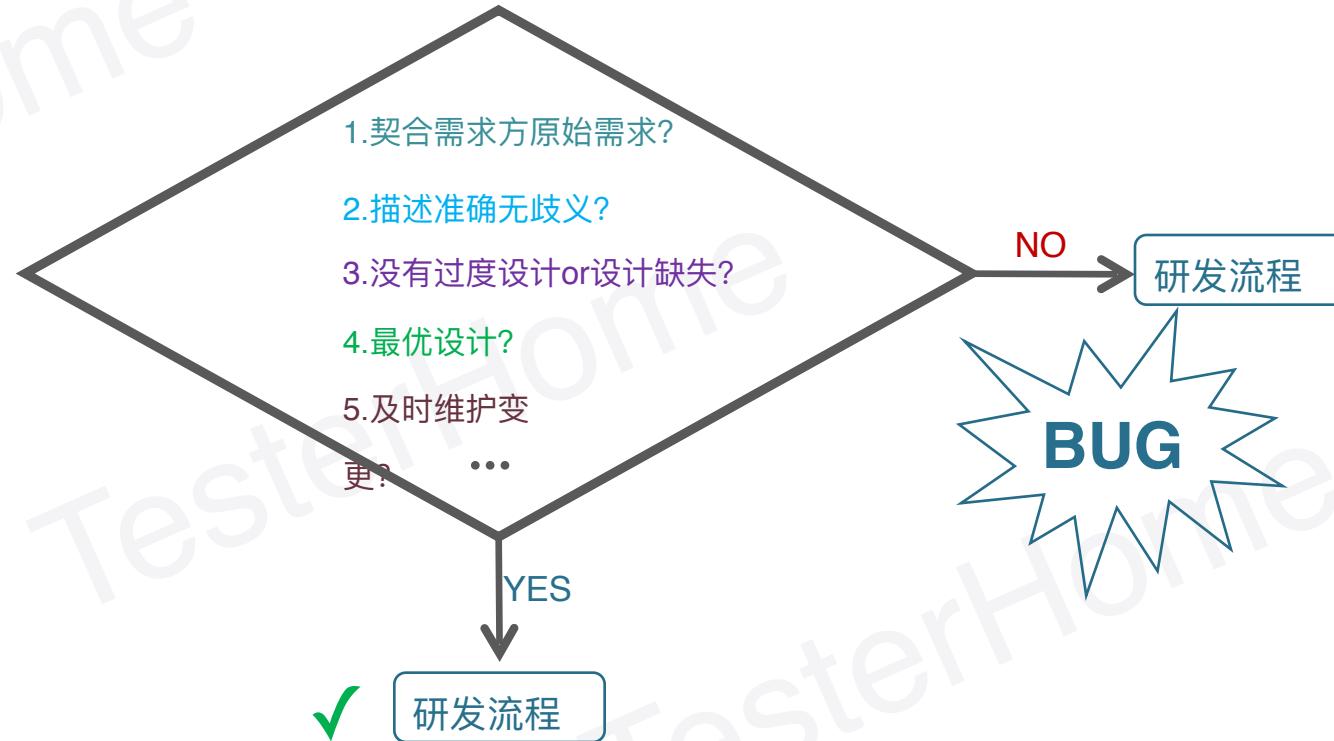
保证需求的原子性，不仅保证了迭代效率，更把缺陷引入的火苗提前熄灭！

缺陷预防实践二：需求测试



- 1 需求评估
- 2 需求拆分
- 3 需求设计
- 4 需求评审

业务/产品



缺陷预防实践二：需求测试

项目实践：

实践环节：需求评审 落实人员：测试人员

要解决的问题：

什么样的需求可以测试？

如何做反向约束？

输出是什么？

VS

遇到的问题：

占用大量时间，变更后，又得重新测试

需求问题各方争执不下，难以定论

对需求测试产生了依赖

缺陷预防实践二：需求测试

MTSC2019

中国移动互联网测试开发大会

实践效果：

需求缺陷					
PDM-1	9	媒体接入	【 <u>客户</u> 】设计问题，见附件	关闭	2019/03/12 2019/04/12
PDM-2	8	媒体接入	新增字段描述不清晰	关闭	2019/04/12
PDM-3	8	【 <u>客户</u> 】账户管理页面，弹窗的确定按钮名称不统一，有时 <u>确定</u> ，需要统一起来	关闭	2019/04/03	2019/04/03
PDM-32	■	【商品设计】 <u>产品设计</u> 用户上传头像的操作，但是原型上却有头像显示	关闭	2019/04/03	2019/04/03
PDM-4	9	媒体广告位	【 <u>广告设计</u> 】规则时，未说明过滤广告位是否可编辑	关闭	2019/02/21 2019/03/19
PDM-5	4	广告位	【 <u>广告设计</u> 】 <u>广告设计</u> 的视频和图标尺寸ID标注有误	待解决	2019/02/19 2019/02/19
PDM-6	3	广告位	【 <u>广告设计</u> 】 <u>35、36</u> 对应的模板名称及预览图有误	关闭	2019/02/14 2019/02/19
PDM-7	54	广告位	【 <u>平台广告主</u> 】增加CPM、ICPM字段，没有明确在总表或详细报表或两者都添加	关闭	2019/02/12 2019/04/01
PDM-8	11	广告位	【 <u>天高点</u> 】 <u>无元</u> ，广告位在于 <u>客户</u> ，每天有预算则取预算，无数据，没有描述清楚是取哪一部分	关闭	2019/01/02 2019/04/01
PDM-9	6	优化	【 <u>优化</u> 】只有 限才要优 ， 击率	关闭	2018/12/29 2019/04/01



找出需求缺陷，输出测试框架，提前发现隐藏问题！

缺陷预防实践三：jira交付规范



1 技术方案设计

2 技术方案评审

技术架构师

1-习惯

边做边想

2-产出

除了代码，
一无所有

BUG

3-影响

追溯？学
习？排
查？



缺陷预防实践三：jira交付规范

项目实践：

实践环节：技术方案评审 落实人员：研发人员



缺陷预防实践三：jira交付规范

实践效果：

The screenshot shows a Jira issue page for a task titled "wap媒体协议接口逻辑测试". The task is marked as "完成" (Completed) with a green status indicator. The "Attachments" section contains four files: "cog接口.jmx" (12 kB), "wap接口.jmx" (13 kB), "wap接口说明.docx" (21 kB), and "讯接说明.docx" (138 kB). A large blue callout bubble on the right side of the page says "jira交付文档".

问题详情

类型: 子任务
优先级: 中
影响版本: 无
模块: 无
标签: 无
Sprint: 2018-03-01 (冲刺 2)

状态: 完成 (查看工作流)
解决结果: 完成
解决版本: 无
安全级别: 项目问题

描述

1、测试wap媒体协议
2、测试cookie mapping服务接口

附件

将文件拖放到此处以添加附件，或者 浏览.

文件名	大小
cog接口.jmx	12 kB
wap接口.jmx	13 kB
wap接口说明.docx	21 kB
讯接说明.docx	138 kB

强制的总结和归纳流程，培养优秀的设计习惯，减少设计中可能隐藏的bug！

缺陷预防实践四：版本质量考核体系



- 1 编码
- 2 代码评审
- 3 开发自测
- 4 版本构建

研发人员

质量不可视，未量化

版本质量优劣不考核

历史质量难追溯

BUG

- 1.质量意识薄弱
 - 2.自我要求松懈
 - 3.竞争氛围低靡
 - 4.缺陷警觉性降低
- ...

缺陷预防实践四：版本质量考核体系

项目实践：

实践环节：开发自测 落实人员：研发人员、测试人员



合理的考核标准

web端/服务端、产品对内/对外、不同的研发模式，需要根据情况制定标准。



弹性拔高标准

当整体质量在提高的时候，需要对考核标准进行一定程度的拔高，才能督促开发人员不断的提高自身对质量的要求，始终敬畏质量。



建立奖惩制度

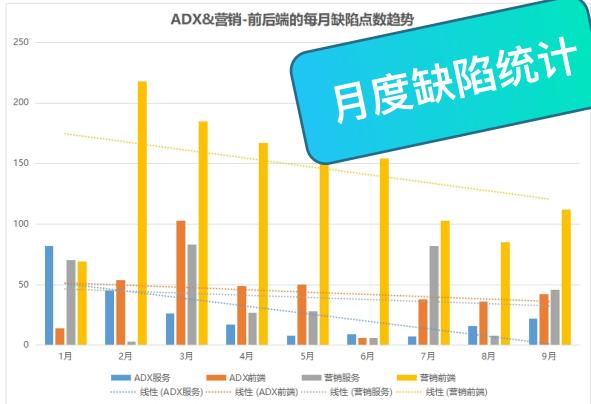
按月汇总，将版本质量考核成绩变现为研发绩效加分，并在团队内公示形成浓郁的质量内建之风。

缺陷预防实践四：版本质量考核体系

实践效果：

附注1：

- a. 本月所有方向均未出现版本质量考核分档为D的版本，期望继续保持。
- b. 本月各方向的缺陷总体引入量要比8月份普遍增长，各方向需要加强重视，将在节后组织详细缺陷复盘。



缺陷快速跟进
在立项会认为你的团队或项目的重要内容
通过缺陷检测和定位。

高质缺陷库
 » 01-需求管理规范
 » 02-测试流程规范
 » 03-测试用例规范
 » 04-测试报告规范
 » 05-版本发布构建规范
 » 06-DH缺陷代号规范
 » 07-WTR缺陷口令规范
 » 08-板卡版本管理规范
 » 09-设备资源考核规范
 » 10-测试报告书规范
 » 11-版本上线规范
 » 12-缺陷管理规范
 » 13-线上运营管理规范
 » 14-外部产品测试规范
 » 15-安全测试规范流程
 » 16-SVN代码使用规范
 » 03-测试环境建设
 » 04-测试云平台
 » 05-测试培训体系
 » 06-营销评估体系
 » 07-销售人员考核
 » 08-技术共享社区

大势缺陷库-2019最新版: 2019-AJ黄纳大势缺陷库质量考核标准

版本质量等级	A	B	C	
1.总工时 = (0,1)	最高为C			
1.总工时 = (1,3)	最高为C	同时满足: “一般缺陷≤1 “低级缺陷≤2 “中级缺陷≤3 “严重缺陷≤0 “低级缺陷=0” “缺陷报告不完整 ”BVT失败	同时满足: “一般缺陷≤2 “低级缺陷≤2 “中级缺陷≤4 “严重缺陷≤1 “低级一般缺陷≤1 “中级严重缺陷≤1 “低级严重缺陷=1 “缺陷报告不完整 ”BVT失败	同时满足: “一般缺陷≤2 “低级缺陷≤2 “中级缺陷≤4 “严重缺陷≤2 “低级一般缺陷≤2 “中级严重缺陷≤1 “低级严重缺陷=1 “缺陷报告不完整 ”BVT失败
2.总工时 = (3,5)	最高为B	同时满足: “一般缺陷=0 “低级缺陷≤1 “中级缺陷≤4 “严重缺陷=0 “低级缺陷=0” “缺陷报告不完整 ”BVT失败	同时满足: “一般缺陷≤3 “低级缺陷≤5 “中级缺陷≤6 “严重缺陷≤1 “低级一般缺陷≤2 “中级严重缺陷≤1 “低级严重缺陷=1 “缺陷报告不完整 ”BVT失败	同时满足: “一般缺陷≤3 “低级缺陷≤5 “中级缺陷≤6 “严重缺陷≤2 “低级一般缺陷≤2 “中级严重缺陷≤2 “低级严重缺陷=1 “缺陷报告不完整 ”BVT失败
3.总工时 = (5,7)		同时满足: “一般缺陷=0 “低级缺陷≤1 “中级缺陷≤3 “严重缺陷=0 “低级缺陷=0” “缺陷报告不完整 ”BVT失败	同时满足: “一般缺陷≤5 “低级缺陷≤7 “中级缺陷≤9 “严重缺陷≤2 “低级一般缺陷≤3 “中级严重缺陷=1 “低级严重缺陷≥2 “缺陷报告不完整 ”BVT失败	同时满足: “一般缺陷≤5 “低级缺陷≤7 “中级缺陷≤9 “严重缺陷≤3 “低级一般缺陷≤3 “中级严重缺陷=2 “低级严重缺陷≥2 “缺陷报告不完整 ”BVT失败

质量考核标准



合理的量化质量、建立奖惩制度，以此督促研发自测，消灭低级缺陷！

缺陷预防实践五：版本发布构建规范



- 1 编码
- 2 代码评审
- 3 开发自测
- 4 版本构建

研发人员



版本内容：漏
写、错写



业务联动：不说
明



影响范围：不
明确

代码提交：
遗漏、多余

误提交一行与版本内容无关的测试代码，造成损失4万元！

项目实践：

实践环节：版本构建

落实人员：研发人员

01

有序的发版制度

串行或者并行，不同的研发模式发版方式不同。

02

规范CI构建说明

目的、开发/设计范围、新增/修改说明、已知问题、联调范围。

03

Diff 代码

重视每一行提交的代码。

实践效果：

1. 组件列表

注明本次要发布哪些组件，帮助研发人员知道自己总结能力差多

2. 构建

写清楚了，这样以后自己总结能力就多

另外：

(1)、不建议一个版本发布多个功能，否则上线后其中一个出了复才能上线。

(2)、不建议多人凑在一起发版本。

以上两个建议执行具体由各组组长根据实际情况定夺。

3. 开发/修改设计范围

注明增、删、改了哪些模块：

新增模块描述清楚功能；

删除模块描述清楚删除原因；

修改模块描述清楚修改了什么功能，从什么变更到什么。

做到能让同组其它研发同学一目了然，知道你做了什么事情；

做到能让测试同学看完非常明确测试范围。

4. 新增/修改说明

注明增、删、改了哪些文件：

CI构建规范

diff 代码

开发/设计范围

1. 老统

2. 修复平台开启标签若获取标签为空报空指针异常的问题

Downplat_ID_00001_IIflyMediaApi.java, 添加请求老逻辑请求到新逻辑请求参数结构不变，并修改了老到新物料转换的方法名称

开发修改说明

- Downplat_ID_00001_IIflyMediaApi.java, 添加请求老逻辑请求到新逻辑请求参数结构不变，并修改了老到新物料转换的方法名称

- Downplat_ID_00001_IIflyMediaApi.java, 修改了老逻辑请求到新逻辑请求参数结构不变，并修改了老逻辑请求到新逻辑请求的物料名

- AdConstants.java, 增加视频格式常量

- CacheUtil.java, 增加根据新逻辑请求到老逻辑请求的映射，解决原生的文字长度的方法

- Upplat_ID_00001_IIflyDsp.java, 修复平台开启标签若获取标签为空报空指针异常的问题

- Upplat_ID_00001_IIflyDsp.java, 兼容老统一 dsp 下发的物料名，修复平台开启标签若获取标签为空报空指针异常的问题，并对视频广告的请求参数构建做了优化

构建报告

新增/修改说明

1. 老统一

2. 修复平台开启标签若获取标签为空报空指针异常的问题

已知问题列表

(已经和产品确认)

1. 因为新媒体协议横幅插屏不支持 HTML 片段，所以不支持老统一 平台下发 HTML 片段给新媒体，如果有 html 片段需求则需要通过自定义的 HTML 片段

2. 因为新媒体协议横幅插屏不支持竖屏式的创意。所以不支持老统一 dsp 平台下发竖屏式的广告给新媒体

3. 不支持广告尺寸视频这种形式，不列入单元测试。限制背景图片尺寸

4. 视频尺寸不能超过 1080*1920，否则会限制

强制而明确的构建list，帮助研发人员总结梳理以进行完整的交付，减少交付过程中引入的缺陷！

缺陷预防实践六：缺陷复盘



测试人员

A 复盘

B 因定位

C 盘总结

缺陷预防实践六：缺陷复盘

实践效果：

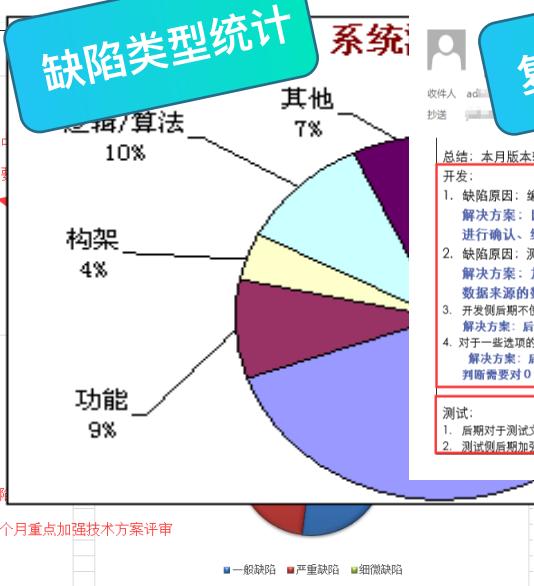
缺陷起因说明

严重性:	严重
发现深度:	正常发现
发生率:	1.无法遂
重现步骤:	v 原因 —ds
发现版本:	adxsm ■ adxsm ■
计划解决版本:	adxsm ■ adxsm ■
解决方法*:	已修复
修复细节:	多样原生 校验，现
缺陷起因:	设计 - 设
迟发现缺陷:	否
验证细节:	验证通过
典型缺陷:	否

人员缺陷统计



缺陷类型统计



复盘总结报告

总结：本月版本较多，但是研发大大的版本质量都比较高，因此本月的缺陷相对较少，为各位点赞！

开发：

1. 缺陷原因：编码问题&设计问题

解决方案：因业务逻辑错误或未按设计实现而引发的问题，开发人员要熟悉需求点，理清业务逻辑，对设计的方案再次进行确认、编码，尽量避免遗漏功能点；对于重构代码时，开发侧须加强自测，以保证功能的正常。

2. 缺陷原因：测试文档

解决方案：加强测试文档编写规范，对于提供的测试文档，自测时也要确认所写的文档是否准确，理清计算逻辑和明确数据来源的数据库表，对于从需求文档或者其他地方直接拷贝的公式要确认是否正确。

3. 开发侧后期不使用HTML拼接方法：

解决方案：后端开发不允许使用HTML拼接，页面渲染由前端处理，避免引起界面直接展示标签问题。

4. 对于一些选项的枚举值不从0开始：

解决方案：后期对于像流量类型0代表APP流量、1代表OTT流量这样枚举值的设置，不要从0开始，因为前端选择0传到后端，后端判断需要对0进行特殊处理。

测试：

1. 后期对于测试文档书写有较为严重错误的测试例将打回重新书写正确后在进行测试

2. 测试侧后期加强自身对业务的理解和对缺陷的界定

从全面的缺陷统计分析，到缺陷复盘总结报告，通过有效的反馈对研发人员进行指导和约束！

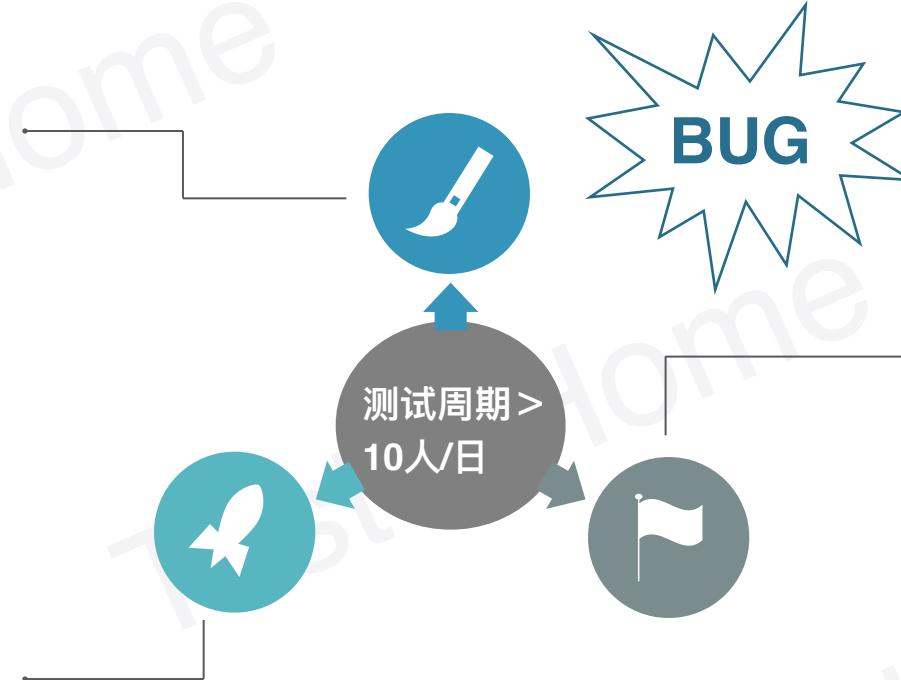
缺陷预防实践七：版本粒度精准切割

自测有边界

自测局限范围在个人负责的部分，导致整体功能有自测盲区

合并风险大

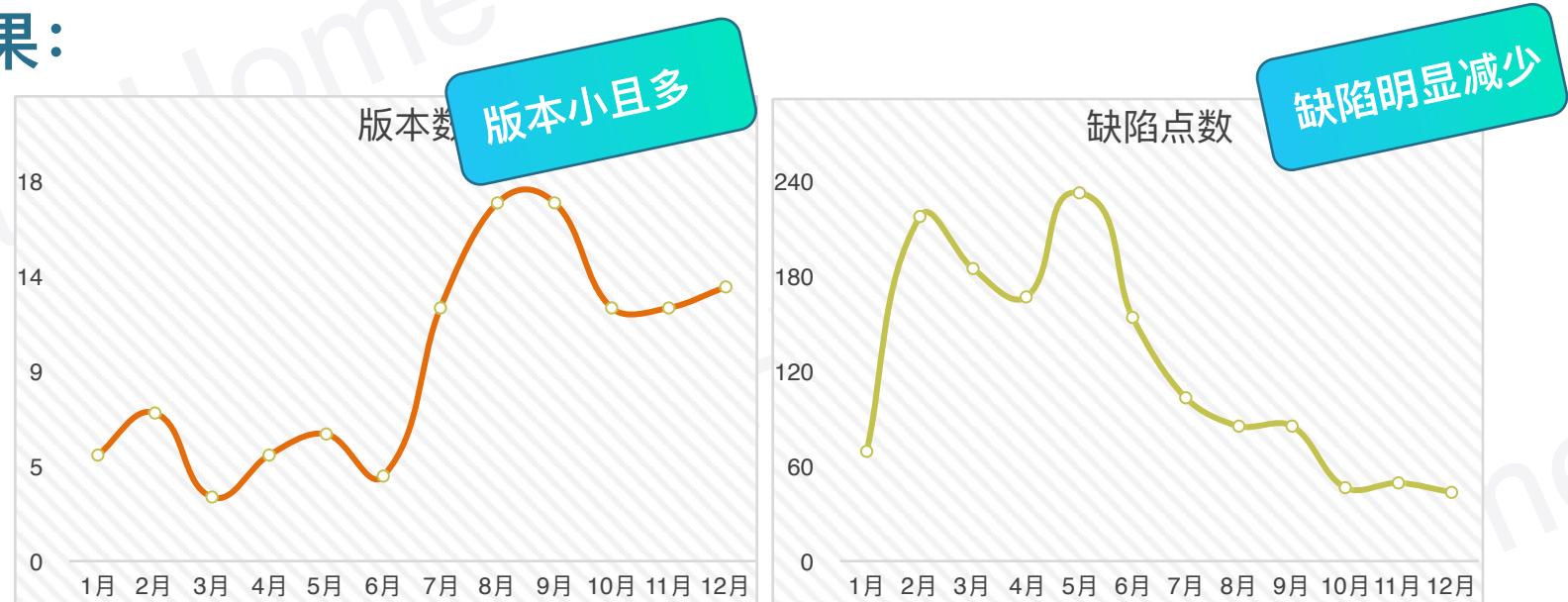
多个开发一起开发的代码，合并时更容易引入风险



影响范围难界定
功能点太多，影响范围难以确定

缺陷预防实践七：版本粒度精准切割

实践效果：



合理的版本粒度，不仅提升效率，更能减少代码耦合带来的缺陷！

5

第五部分

总结展望

总结展望



从问题驱动，到价值驱动！

Q&A

Thanks

!

孙玉

科大讯飞AI营销业务群测试架构师

